Contenido

Introducción 2
Requerimientos 2
Requerimiento #1: IDE Eclipse 2
Requerimiento #2: Apache Tomcat, Jboss y MySQL 2
Desarrollo 3
Paso #1: Ejecutar Eclipse 3
Paso #2: WorkSpace 3
Paso #3: Dynamic Web Project 3
Paso #4: Nombre del Proyecto 4
Paso #5: Configuración JBoss o Apache Tomcat 4
Paso #6: Servidores 4
Paso #7: JDBC 5
Paso #8: Base de Datos en MySQL 5
Paso #9: AltaContacto.jsp 5
Paso #10: listaContactos.jsp 6
Paso #11: eliminarContacto.jsp 7
Paso #12: editarContactos.jsp 8
Paso #13: Correr paginas JSP 10

Introducción

El presente manual explica de forma clara mediante pasos sencillos y visualmente mediante imágenes, la utilización de ”JSP” mediante “Eclipse” para manipular bases de datos con “MySQL” con “Apache Tomcat” o “jBoss”.

Requerimientos

Antes de empezar con la explicación sobre la configuración de Eclipse y las herramientas “Apache Tomcat” y “jBoss”, debes de tomar en cuenta los siguientes requerimientos antes de comenzar:

Requerimiento #1: IDE Eclipse

Actualizar la Maquina Virtual de Java e instalar el IDE Eclipse:

Requerimiento #2: Apache Tomcat, Jboss y MySQL

Sobre “JBoss” y “Apache Tomcat”, se recomienda instalar solo un servidor para evitar conflictos con el puerto 8080, y configurarlo para que su activación sea “Automática”.

	Apache Tomcat 5.5 ➢ Página oficial: http://tomcat.apache.org/ ➢ Puedes descargarlo aquí: http://www.devtroce.com/link-externo/http://tomcat.apache.org/download-60.cgi
	Jboss-4.2.0.GA ➢ Página oficial: http://www.jboss.org/jbossas/downloads/

MySQL	MySQL • Instalado (Checar Manual de Instalación Anexo a este Manual) • Página oficial: http://www.mysql.com/ • Al instalarlo verifica que utilice el puerto 3306 y escribe el password para conectarte como en el ejemplo: `String strcon = "jdbc:mysql://localhost:3306/agenda?user=root&password=admin";`

Desarrollo

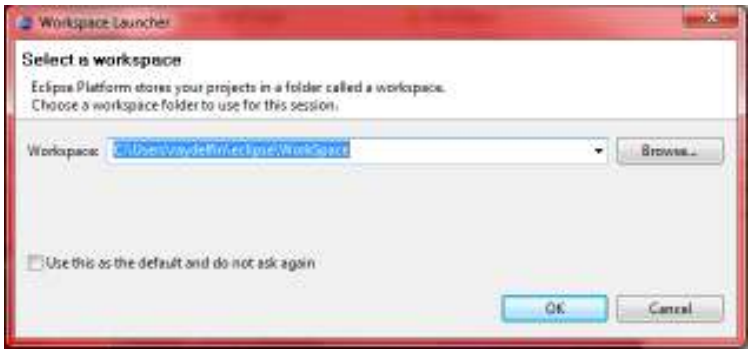
Una vez cumplidos los requisitos previos sigue los siguientes pasos al pie de la letra:

Paso #1: Ejecutar Eclipse

Abre la aplicación "Eclipse".

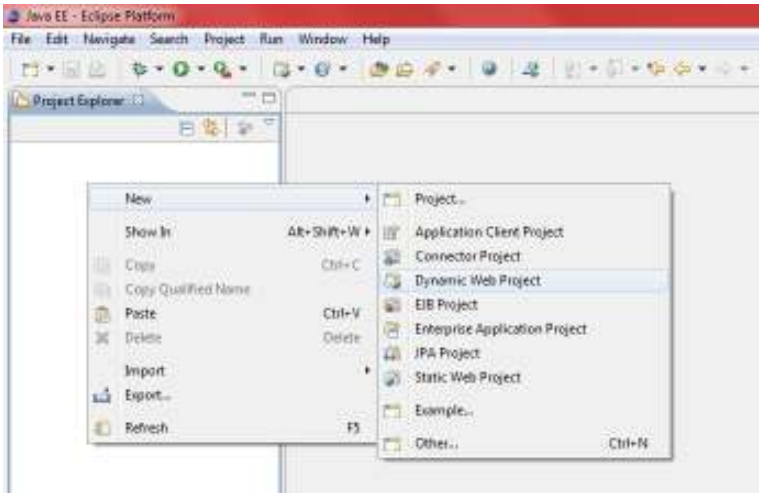
Paso #2: WorkSpace

Selecciona el "WorkSpace" donde desees realizar tu aplicación.


Workspace Launcher
Select a workspace
Eclipse Platform stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.
Workspace:
Browse...
Use this as the default and do not ask again
OK
Cancel

Paso #3: Dynamic Web Project

Una vez dentro de Eclipse, haz clic derecho sobre la ventana "Project Explorer", selecciona "New" y después "Dynamic Web Project".



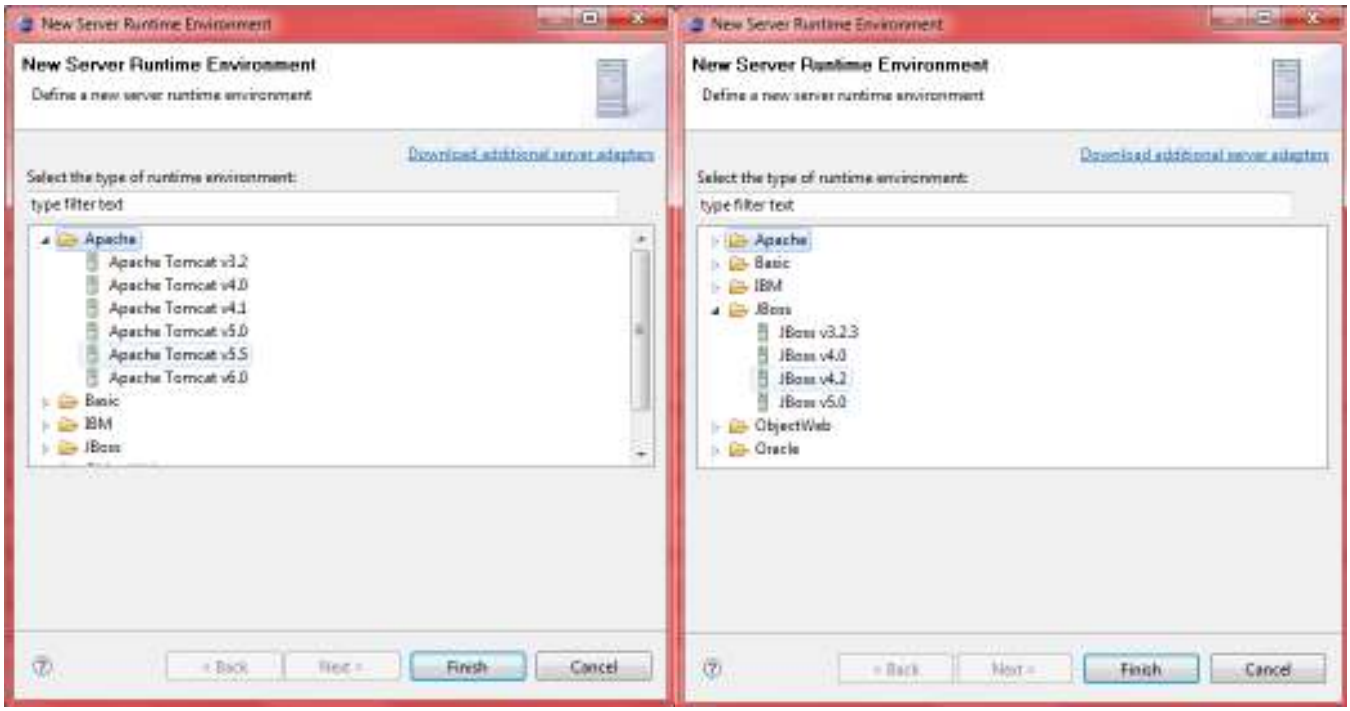
Paso #4: Nombre del Proyecto

A continuación se muestra la siguiente interfaz donde debes de teclear el nombre del proyecto "Agenda", luego haz clic en el botón "New" del área "Target Runtime".

Paso #5: Configuración JBoss o Apache Tomcat

Se mostrara la siguiente interfaz, selecciona "Apache Tomcat v5.5" o "Jboss v4.2" y presiona en "Finish".

Nota: Una vez seleccionado el tipo de servidor, la primera debes introducir la ruta donde lo instalaste.



Paso #6: Servidores

Una vez seleccionado alguno de los servidores de aplicaciones, presiona el botón "Finish" en la interfaz del Paso#4.

Paso #7: JDBC

Una vez hecho esto, necesitaremos el driver **JDBC** para conectarnos a MySQL, la cual pueden descargar desde el sitio oficial http://dev.mysql.com/downloads/connector/j/ , al descargar el archivo (.zip), debemos copiarlo en el siguiente path dentro de nuestro proyecto (dentro del workspace):

..\Agenda\WebContent\WEB-INF\lib\mysql-connector-java-5.1.10-bin.jar

Paso #8: Base de Datos en MySQL

Una vez instalado el driver JDBC, ya tenemos listo nuestro proyecto para crear las paginas JSP, en el presente ejemplo "Agenda", debemos crear nuestra base de datos en MySQL:

- CREATE DATABASE agenda;
- USE agenda;
- CREATE TABLE contacto(ID int, Nombre varchar(50), Dir varchar(100), Tel varchar(30));

Paso #9: AltaContacto.jsp

Ya creada nuestra base de datos en MySQL, crearemos las paginas JSP necesarias para manipular nuestra "Agenda". Haz clic sobre el proyecto Agenda → New → JSP, y guárdalo como "AltaContacto.jsp", al desplegarse el código del jsp, introduce dentro de las etiquetas <body> </body>, el siguiente código:

```
<%@ page import="java.io.*,java.util.*,java.net.*,java.sql.*" %>
<%
	if(request.getParameter("GRABAR") != null)
	{	// objetos de enlace
		Connection canal = null;
		ResultSet tabla= null;
		Statement instruccion=null;
		String strcon = "jdbc:mysql://localhost:3306/agenda?user=root&password=admin";
		// abriendo canal o enlace en su propio try-catch
		try
		{	Class.forName("com.mysql.jdbc.Driver").newInstance();
			canal=DriverManager.getConnection(strcon);
			instruccion = canal.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
ResultSet.CONCUR_UPDATABLE);
		}
		catch(java.lang.ClassNotFoundException e){}
		catch(SQLException e) {};

		//cargando los campos a grabar
		int id = Integer.parseInt(request.getParameter("id"));
		String nombre = request.getParameter("nombre");
		String dir = request.getParameter("dir");
		String tel = request.getParameter("tel");
		String cadSQL = "insert into contacto(ID, Nombre, Dir, Tel) values(" + id + ", '" + nombre + "', '" + dir + "', '" +
tel + "');";

		try
		{	// agregando renglon (insert)
			int n=instruccion.executeUpdate(cadSQL);
			//avisando que se hizo la instruccion
			out.println("REGISTRO INSERTADO");
		}
```

```
                catch(SQLException e)
                {out.println(e);};
                try
                {       // tabla.close();
                        instruccion.close();
                        canal.close();
                }
                catch(SQLException e)
                {out.println(e);};
        };
%>
        <FORM ACTION=AltaContacto.jsp METHOD=post>
        <table width="34%" border="10" align="center" cellpadding="1" cellspacing="1" bordercolor="#000099"
bgcolor="#FFFFFF">
          <tr bordercolor="#FFFFFF" bgcolor="#FFFFFF">
            <th>ID</th>
            <th><div align="left"><input type="text" name="id"  size="10" "> </div>
            </th>
          </tr>
          <tr bordercolor="#FFFFFF" bgcolor="#FFFFFF">
            <th> </th>
            <th> <div align="left"></div></th>
          </tr>
          <tr bordercolor="#FFFFFF" bgcolor="#FFFFFF">
            <th>Nombre</th>
            <th><div align="left"><input type="text" name="nombre" size="40" ></div>
            </th>
          </tr>
          <tr bordercolor="#FFFFFF">
            <th>Dirección</th>
            <th> <div align="left"><input type="text" name="dir" size="40" ></div>
            </th>
          </tr>
          <tr bordercolor="#FFFFFF">
            <th>Tel</th>
            <th> <div align="left"><input type="text" name="tel" size="40" ></div>
            </th>
          </tr>
          <tr bordercolor="#FFFFFF">
            <th></th>
            <th><input type="submit" name="GRABAR" value="Insertar Contacto"></th>
          </tr>
</table>
</FORM>
```

Paso #10: listaContactos.jsp

Checa que en el código anterior empieza con los caracteres <%, y termina en %>, todo lo que se encuentre dentro de estas llaves debe de ser código jsp. El siguiente código es para visualizar los contactos que demos de alta, haz clic sobre el proyecto Agenda → New → JSP, y guárdalo como "listaContactos.jsp", al desplegarse el código del jsp, introduce dentro de las etiquetas <body> </body>, el siguiente código:

```
<%@ page import="java.io.*,java.util.*,java.net.*,java.sql.*" %>
<%
        // declarando y creando objetos globales
        Connection canal = null;
```

```
ResultSet tabla= null;
Statement instruccion=null;
String strcon = "jdbc:mysql://localhost:3306/agenda?user=root&password=admin";
if(request.getParameter("OK") != null)
{	// abriendo canal o enlace en su propio try-catch
	try
	{	Class.forName("com.mysql.jdbc.Driver").newInstance();
		canal=DriverManager.getConnection(strcon);
		instruccion = canal.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
		ResultSet.CONCUR_UPDATABLE);
	}
	catch(java.lang.ClassNotFoundException e){} catch(SQLException e) {};
	//leyendo tabla en disco y pasandola al resultset
	try
	{	tabla = instruccion.executeQuery("select * from contacto");
		// mandando resultset a una tabla html
		out.println("<TABLE Border=10 CellPadding=5><TR>");
		out.println("<th>ID</th> <th>Nombre</th> <th>Dir</th> <th>Tel</th></TR>");
		// ciclo de lectura del resultset
		while(tabla.next())
		{	out.println("<TR>");
			out.println("<TD>"+tabla.getString(1)+"</TD>");
			out.println("<TD>"+tabla.getString(2)+"</TD>");
			out.println("<TD>"+tabla.getString(3)+"</TD>");
			out.println("<TD>"+tabla.getString(4)+"</TD>");
			out.println("</TR>");
		}; // fin while
		out.println("</TABLE></CENTER></DIV></HTML>");
		// cerrando resultset
		tabla.close(); instruccion.close();canal.close();
	} //fin try no usar ; al final de dos o mas catchs
	catch(SQLException e) {};
};
// construyendo forma dinamica
out.println("<FORM ACTION=listaContactos.jsp METHOD=post>");
out.println("<INPUT TYPE=SUBMIT NAME=OK VALUE=CONSULTA><BR>");
out.println("</FORM>");
%>
```

Paso #11: eliminarContacto.jsp

Haz clic sobre el proyecto Agenda → New → JSP, y guárdalo como “eliminarContacto.jsp”, al desplegarse el código del jsp, introduce dentro de las etiquetas <body> </body>, el siguiente código:

```
<%@ page import="java.io.*,java.util.*,java.net.*,java.sql.*" %>
<%
	if(request.getParameter("eliminar") != null)
	{	// objetos de enlace
		Connection canal = null;
		ResultSet tabla= null;
		Statement instruccion=null;
		String strcon = "jdbc:mysql://localhost:3306/agenda?user=root&password=admin";
		// abriendo canal o enlace en su propio try-catch
		try
		{	Class.forName("com.mysql.jdbc.Driver").newInstance();
```

```
                        canal=DriverManager.getConnection(strcon);
                        instruccion = canal.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
ResultSet.CONCUR_UPDATABLE);
                }
                catch(java.lang.ClassNotFoundException e){}
                catch(SQLException e) {};
                //cargando los campos a grabar
                int id = Integer.parseInt(request.getParameter("id"));
                String cadSQL = "DELETE FROM contacto WHERE ID = " + id + ";";
                try
                {       // agregando renglon (insert)
                        int n=instruccion.executeUpdate(cadSQL);
                        //avisando que se hizo la instruccion
                        out.println("REGISTRO ELIMINADO");
                }
                catch(SQLException e)
                {out.println(e);};
                try
                {       // tabla.close();
                        instruccion.close();
                        canal.close();
                }
                catch(SQLException e)
                {out.println(e);};
        };
%>
        <FORM ACTION=eliminarContacto.jsp METHOD=post>
        <table width="34%" border="10" align="center" cellpadding="1" cellspacing="1" bordercolor="#000099"
bgcolor="#FFFFFF">
          <tr bordercolor="#FFFFFF" bgcolor="#FFFFFF">
            <th>ID</th>
            <th><div align="left"><input type="text" name="id"  size="10" "> </div>
            </th>
          </tr>
          <tr bordercolor="#FFFFFF">
            <th></th>
            <th><input type="submit" name="eliminar" value="Eliminar Contacto"></th>
          </tr>
</table>
</FORM>
```

Paso #12: editarContactos.jsp

Haz clic sobre el proyecto Agenda → New → JSP, y guárdalo como "editarContactos.jsp", al desplegarse el código del jsp, introduce dentro de las etiquetas <body> </body>, el siguiente código:

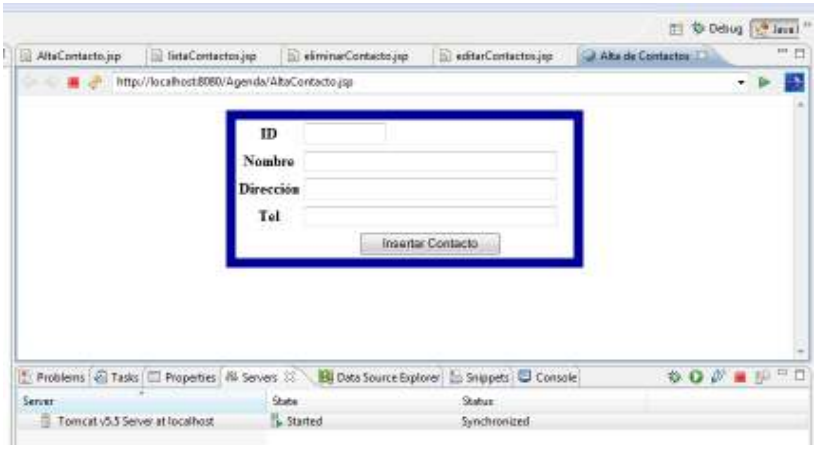
```
<%@ page import="java.io.*, java.util.*, java.net.*, java.sql.*" %>
<%! int id = 0; %>
<%
        Connection canal = null;
        ResultSet tabla= null;
        Statement instruccion=null;
        String strcon = "jdbc:mysql://localhost:3306/agenda?user=root&password=admin";
        String cadSQL = "";
        // codigo del evento BUSQUEDA y recordar construir una nueva forma dinamica
        if(request.getParameter("buscar") != null)
```

```
        {       try
                {       Class.forName("com.mysql.jdbc.Driver").newInstance();
                        canal=DriverManager.getConnection(strcon);
                        instruccion = canal.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
                        ResultSet.CONCUR_UPDATABLE);
                }
                catch(java.lang.ClassNotFoundException e){}
                catch(SQLException e) {};
                id = Integer.parseInt(request.getParameter("id2"));
                cadSQL="SELECT Nombre, Dir, Tel FROM contacto WHERE ID= " + id + ";";
                try
                {       tabla = instruccion.executeQuery(cadSQL);
                        tabla.next();
                        out.println("<FORM ACTION=editarContactos.jsp METHOD=POST>");
                        out.println("<table width=34% border=10 align=center cellpadding=1 cellspacing=1
bordercolor=#000099 bgcolor=#FFFFFF>");
                        out.println("<tr bordercolor=#FFFFFF bgcolor=#FFFFFF><th>ID</th>");
                        out.println("<th><div align=left><input type=\"text\" name=\"id\" disabled value=\"" + id + "\"
size=40> </div> </th> </tr>");
                        out.println("<tr bordercolor=#FFFFFF bgcolor=#FFFFFF><th>Nombre</th>");
                        out.println("<th><div align=left><input type=\"text\" name=\"nombre\" value=\"" + tabla.getString(1) +
"\" size=40> </div> </th> </tr>");
                        out.println("<tr bordercolor=#FFFFFF bgcolor=#FFFFFF><th>Dirección</th>");
                        out.println("<th><div align=left><input type=\"text\" name=\"dir\" value=\"" + tabla.getString(2) + "\"
size=40> </div> </th> </tr>");
                        out.println("<tr bordercolor=#FFFFFF bgcolor=#FFFFFF><th>Tel</th>");
                        out.println("<th><div align=left><input type=\"text\" name=\"tel\" value=\"" + tabla.getString(3) + "\"
size=40> </div> </th> </tr>");
                        out.println("<tr bordercolor=#FFFFFF> <th></th>    <th><input type=\"submit\" name=\"editar\"
value=\"Editar Contacto\"></th> </tr>");
                        out.println("</table></FORM>");
                        tabla.close();instruccion.close();canal.close();
                }
                catch(SQLException e) {}
                catch(Exception ex){};
        }; // fin evento buscar
        // codigo de evento EDICION
        String temp2=request.getParameter("editar");
        if(temp2==null)temp2=" ";
        if(request.getParameter("editar") != null)
        {   id = Integer.parseInt(request.getParameter("id"));
                String nombre = request.getParameter("nombre");
                String dir = request.getParameter("dir");
                String tel = request.getParameter("tel");
        try
        {       Class.forName("com.mysql.jdbc.Driver").newInstance();
                canal=DriverManager.getConnection(strcon);
                instruccion = canal.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
                ResultSet.CONCUR_UPDATABLE);
        } catch(java.lang.ClassNotFoundException e){} catch(SQLException e) {};
        try
        {       tabla = instruccion.executeQuery("SELECT * FROM contacto;");
        } catch(SQLException e) {};
        cadSQL = "UPDATE contacto SET NOMBRE='"+ nombre + "', Dir='"+ dir +"', Tel='" + tel + "' WHERE ID =" + id +";";
        try{instruccion.executeUpdate(cadSQL); }catch(SQLException e) {};
        try {tabla.close();instruccion.close();canal.close();} catch(SQLException e) {};
```

```
        out.println("CONTACTO EDITADO");
        }; // fin evento editar
        // construyendo forma dinamica
        out.println("<FORM ACTION=editarContactos.jsp METHOD=post>");
        out.println("<table width=\"34%\" border=\"10\" align=\"center\" cellpadding=\"1\" cellspacing=\"1\"
bordercolor=\"#000099\" bgcolor=\"#FFFFFF\">");
        out.println("<tr bordercolor=\"#FFFFFF\" bgcolor=\"#FFFFFF\"> <th>ID del Contacto a Editar:</th>");
        out.println("<th> <div align=\"left\"> <input type=\"text\" name=\"id2\" size=\"10\"> </div></th></tr>");
        out.println("<tr> <th></th> <th><input type=\"submit\" name=\"buscar\" value=\"Buscar Contacto\"></th>");
        out.println("</table></FORM>");
%>
```

Paso #13: Correr paginas JSP

A continuación selecciona “AltaContacto.jsp”, haz clic en “Run as” → “Run on server”, para visualizar la siguiente interfaz:



Para visualizar las demás interfaces, de la misma manera haz clic en “Run as” → “Run on server” para visualizar la pagina. Por ejemplo listaContactos.jsp: